School of Computer Science Engineering and Information Systems

Department of Computer Applications

Fall Semester 2023-24
ITA6099 – Master Thesis
Second Review Report

Reg. No: 22MCA0195

Name: Aakash Jaiswal

Guide Name: Dr. Jagadeesan S

Master Thesis Title: An Alarm Based Real Time Face Mask Detection and Attendance System for Manufacturing Plants

Programming Language Used for Implementation: Python

Framework Used: Flask

Used for Implementation: Pycharm

Dataset Used for Implementation: Kaggle Website

Sl no.	Contents	Page no.
1.	Abstract	**1**
2.	Introduction	**2**
3.	Related Works	**3 - 10**
4.	Problem Definition	**11**
5.	Architecture Diagram	**12 - 14**
6.	Flow Diagram	**15**
7.	Module Description	**17**
8.	Sample codes	**18 - 23**
9.	Sample test cases & Test Cases	**23**
10.	Performance Metrics	**24 - 26**
11.	Conclusion & Future Enhancement	**27**
12.	References	**28 - 30**

List Of Figures

Figure No.	Title	Page No.
1	Architecture Diagram for Face Mask Detection System	12
2	Architecture Diagram for Face Recognition Attendance System	14
3	Flow-chart for Face Mask detection System	15
4	Flow-chart for Face Recognition Attendance System	15
5	Input output of the face mask detection system	23
6	Performance metric of face mask detection system	24
7	Training Loss & Accuracy Graph	25

List Of Tables

Table No.	Title	Page No.
1	Training Model on SGD Optimizer	16
2	Training Model on Adam Optimizer	16
3	Hyper Parameters used in Training	17
4	Comparison Table of different models	26

Abstract

Manufacturing Plants are a great source of income for entrepreneurs starting new businesses. A company produces its own products or provides services to other businesses. These factories are located in areas with significant consumer demand for the goods made there and low labour costs. It frequently involves a large time and resource investment in addition to high capital requirements. The output of a manufacturing plant, however, might be quite profitable. Some industrial businesses can generate millions of dollars a year in revenue. Numerous nations have shut down their ports, airports, and domestic transportation systems in response to COVID-19, causing a disruption in both business and civil life. This immediately put an end to many commercial activities around the world. Global manufacturing operations were hampered by the lockdown in India. India took particular precautions to contain the spread by enforcing one of the world's longest lockdowns in order to mitigate the limited supply of medical supplies. The scope of the lockdown disrupted the whole economy, having an impact on both supply networks and manufacturing operations. The biggest problem was labour. The government then took the decision to allow labour to come back to work in manufacturing plants but they set a limit. Here Labour also needs money to feed their family and to do so they need to back to work. I came to know about existing projects where the system only detects face mask and doesn't checks where labour is wearing the mask properly or not. Just think in the absence of a security guard, the labour is going inside to work without wearing a mask. Manually it is very tough to take attendance of those labourers. Our proposed system comes up with an idea to solve these issues. If no mask is detected or not properly worn, it will generate an alarm sound. It also introduces the concept of using facial recognition technology for attendance tracking purposes. It highlights the shortcomings of conventional methods like manual attendance registers and swipe cards, paving the way for an innovative approach.

Keywords: Face Mask Detection; video sequence; CNN ;Image Training; Mobile-net V2; Flask

Introduction

A new strain of the COVID-causing virus, SARS-CoV-2, is responsible for the recent rise in illnesses this winter, although it does not appear to produce more severe disease. It's winter, the season of toasty fires, indoor festivities, and a wave of respiratory sickness. Nearly four years after the epidemic began, people are tired of dealing with it, but the virus is not finished with us. Nationally, a substantial increase in emergency room visits and hospitalizations for covid-19, influenza, and respiratory syncytial virus, or RSV, began in mid-December and looks to be accelerating. Don't forget that there are other dangerous bugs circulating around. More than 20,000 individuals were hospitalised for influenza during the week ending December 30, and the CDC says that RSV levels remain high in several places.

In an era where safety concerns and workforce management are paramount, manufacturing plants face the imperative to innovate their operational protocols. The convergence of advanced technologies offers a promising solution to address these challenges. Our project stands at the forefront of this innovation, proposing two systems, one is real-time face mask detection system and another is face recognition attendance management system in manufacturing plants.

The ongoing COVID-19 pandemic has underscored the importance of protective measures, including the use of face masks, to mitigate the risk of viral transmission. Manufacturing plants, characterised by high-density workspaces and shared facilities, present unique challenges in enforcing mask-wearing compliance among employees. Traditional methods of monitoring and ensuring adherence to safety protocols often prove inadequate and resource-intensive. To address these challenges, our project leverages cutting-edge computer vision algorithms to enable real-time detection of face masks within the manufacturing plant premises. By deploying advanced image processing techniques, our system can accurately identify individuals not wearing masks and trigger immediate alerts. Through facial recognition algorithms, the system can identify and authenticate employees, automating the attendance tracking process with unparalleled accuracy and efficiency. This integration not only enhances security measures but also simplifies administrative tasks, saving valuable time and resources for manufacturing plant managers. By embracing this innovative approach, manufacturing facilities can significantly enhance workplace safety, operational efficiency, and overall productivity in the face of evolving challenges.

Related Works:

[1]Deep Learning Framework to Detect Face Masks from Video Footage

- Proposal of a deep learning approach for detecting facial masks in videos, addressing the challenge of occlusions to face detection algorithms caused by masked regions.
- Utilisation of the MTCNN face detection model for identifying faces and their corresponding facial landmarks, followed by processing with a neoteric classifier leveraging MobileNetV2 architecture as an object detector for masked region identification.
- Testing of the proposed framework on a dataset comprising videos of people in public spaces adhering to COVID-19 safety protocols, showcasing high precision, recall, and accuracy in detecting facial masks.
- Contextualization of the proposed methodology within existing research on facial mask detection and deep learning techniques for video analysis.
- Highlighting the effectiveness of the proposed project compared to existing methods, emphasising its ability to handle occlusions caused by facial masks and achieve superior performance metrics.

[2]TransRPPG: Remote Photoplethysmography Transformer for 3D Mask Face Presentation Attack Detection

- Recent advancements in face recognition system security involve addressing 3D mask face presentation attacks through techniques such as remote photoplethysmography (rPPG), which detects intrinsic liveness clues without relying on mask appearance.
- Despite the potential of rPPG for 3D mask presentation attack detection (PAD), manually designing rPPG features requires expert knowledge, hindering progress in the era of deep learning and big data.
- The proposed TransRPPG framework aims to address this limitation by introducing a pure rPPG transformer that efficiently learns intrinsic liveness representation.
- TransRPPG constructs rPPG-based multi-scale spatial-temporal maps (MSTmaps) from facial skin and background regions and utilizes a transformer to fully exploit global relationships within these maps for liveness representation.
- Comprehensive experiments on benchmark datasets demonstrate the effectiveness of TransRPPG for 3D mask PAD, highlighting its lightweight and efficient nature, making it promising for mobile-level applications.

[3]Face Mask Assistant: Detection of Face Mask Service Stage Based on Mobile Phone

- The proposed detection system leverages mobile phone technology to address the challenge of identifying the service stage of face masks, which is crucial for effective COVID-19 mitigation.
- The system utilises features extracted from micro-photos of face masks and employs the K Nearest Neighbour (KNN) algorithm to achieve a three-result detection system, categorising masks into different usage types.

- Validation experiments indicate promising results, with the system achieving an accuracy of 82.87% on the testing dataset, demonstrating its potential effectiveness.
- The precision and recall rates for different mask types show strong performance, particularly in identifying 'normal use' and 'not recommended' scenarios.
- Future work aims to expand the detection capabilities to encompass more mask types, indicating a commitment to ongoing improvement and adaptation to evolving needs in the fight against COVID-19.

[4]Hybrid Transfer Learning and Broad Learning System for Wearing Mask Detection in the COVID-19 Era

- Existing face detection algorithms do not directly address the need for detecting people wearing masks in the context of COVID-19, necessitating the development of specialised monitoring instruments.
- The proposed approach utilises a two-stage method employing hybrid machine learning techniques, with the first stage employing Faster_RCNN and InceptionV2 structures to detect candidate mask regions, and the second stage using a broad learning system to verify the presence of real facial masks.
- A dataset for wearing mask detection (WMD) comprising 7804 realistic images, with 26403 instances of wearing masks, has been introduced and made available for research purposes.
- Experimental results demonstrate the effectiveness of the proposed approach, achieving an overall accuracy of 97.32% for simple scenes and 91.13% for complex scenes, outperforming compared methods.
- The proposed project addresses the gap in existing face detection algorithms by specifically targeting the detection of individuals wearing masks, thereby contributing to efforts to mitigate the spread of COVID-19 in public places such as hospitals and airports.

[5]Contrastive Context-Aware Learning for 3D High-Fidelity Mask Face Presentation Attack Detection

- Existing 3D mask presentation attack detection (PAD) benchmarks have limitations such as a small number of mask identities, types of sensors, and total video counts, as well as low-fidelity quality of facial masks.
- Basic deep models and remote photoplethysmography (rPPG) methods have achieved acceptable performance on these benchmarks but fall short of meeting the needs of practical scenarios.
- To address these gaps, a large-scale HiFiMask dataset is introduced, containing 54,600 videos recorded from 75 subjects with 225 realistic masks by 7 new kinds of sensors.
- Alongside the dataset, a novel Contrastive Context-aware Learning (CCL) framework is proposed for supervised PAD tasks, leveraging rich contexts accurately among pairs of live faces and high-fidelity mask attacks.

- Experimental evaluations on HiFiMask and three additional 3D mask datasets demonstrate the effectiveness of the proposed method, which aims to bridge the gap to real-world applications and offer improvements over existing approaches.

[6]Edge Deployment Framework of GuardBot for Optimised Face Mask Recognition With Real-Time Inference Using Deep Learning

- Deep learning models on edge devices are gaining attention for AI applications, but deploying them efficiently remains challenging due to computation and memory constraints.
- The paper proposes a framework for the service robot "GuardBot" powered by Jetson Xavier NX, focusing on real-time face mask recognition to combat COVID-19.
-
- The framework employs a dual-stage architecture with CNNs, including MTCNN for face detection and various transfer learning-based custom models for mask classification.
- TensorRT is utilised for model optimization, aiming to improve inference speed on Jetson Xavier NX.
- Compared to existing models, the proposed CNN model for face mask recognition achieves superior accuracy, throughput, and latency after optimization, making it suitable for deployment on mobile robots in various environments.

[7]A Real-Time CNN-Based Lightweight Mobile Masked Face Recognition System

- Existing research acknowledges the challenges posed by the Covid-19 pandemic, particularly regarding the need for effective masked face recognition to enforce safety measures.
- Traditional biometric authentication systems, such as facial recognition, face challenges in the pandemic era due to widespread mask usage, necessitating innovative solutions.
- Prior studies have predominantly focused on deep learning models for masked face recognition, but are limited by the scarcity of suitable datasets containing masked faces.
- The proposed study aims to address this gap by developing a real-time masked detection service and face recognition mobile application, utilising an ensemble of fine-tuned lightweight deep Convolutional Neural Networks (CNN).
- Through experiments on newly constructed datasets, the proposed system demonstrates significant improvements in masked face recognition performance compared to existing state-of-the-art approaches, achieving a validation accuracy of 90.40% with a dataset of 12 individuals and 1849 face samples.

[8]DFFMD: A Deepfake Face Mask Dataset for Infectious Disease Era With Deepfake Detection Algorithms

- Deepfake technology creates fake images and videos by replacing or synthesising faces, posing concerns for various social issues including political manipulation, dissemination of false information, and digital harassment.
- The popularity of face masks due to the infectious disease outbreak has made it more challenging to identify fake videos, increasing the need for advanced Deepfake detection technology.
- The proposed research introduces a Deepfake Face Mask Dataset (DFFMD) and a novel detection model based on Inception-ResNet-v2 with preprocessing stages, feature-based techniques, residual connection, and batch normalisation.
- Compared to traditional methods such as InceptionResNetV2 and VGG19, the proposed model achieves significantly higher accuracy in detecting face-mask-Deep Fakes, reaching 99.81% accuracy.
- Future work should focus on further improving the detection accuracy of Deepfakes with face masks through experimental research.

[9]Mask Attack Detection Using Vascular-Weighted Motion-Robust rPPG Signals

- Existing research highlights the challenge of detecting 3D mask attacks in face recognition systems due to differences in remote photoplethysmography (rPPG) signals between genuine faces and masks.
- Current rPPG-based face anti-spoofing methods face performance degradation due to issues like unstable face alignment and weak rPPG signals.
- The proposed method aims to enhance rPPG signal robustly by employing a landmark-anchored face stitching technique, utilising SIFT keypoints and facial landmarks for precise alignment at the pixel-wise level.
- Additionally, a weighted spatial-temporal representation is introduced to better encode the rPPG signal, focusing on face regions with rich blood vessels, and different colour spaces' characteristics are leveraged.
- The proposed lightweight EfficientNet with a Gated Recurrent Unit (GRU) is designed to extract spatial and temporal features for classification, leading to significant and consistent performance improvements over existing state-of-the-art rPPG-based methods for face spoofing detection.

[10]Mask Detection and Classification in Thermal Face Images

- Thermal imaging is explored as a means to automatically detect the presence and type of face masks, crucial for mitigating virus transmission, especially SARS-CoV-2.
- A dataset of annotated thermal images is extended to include mask type and location information, facilitating deep learning model training.
- The Yolov5 model, particularly in its "nano" version, emerges as the top performer for face mask detection, achieving a mean Average Precision (mAP) exceeding 97% and 95% precision.

- Mask type classification sees success with a convolutional neural network (CNN) model based on an autoencoder initially trained for thermal image reconstruction, yielding an accuracy of 91%.
- The proposed project advances existing research by leveraging thermal imaging for precise mask detection and classification, outperforming previous methods and addressing the pressing need for effective virus transmission control measures.

[11]Faster Region-based Convolutional Neural Network for Mask Face Detection

- Faster R-CNN utilised for masked face detection with three classes: no mask, incorrectly masked, and correctly masked faces.
- Two-stage approach involves Region Proposal Network (RPN) for candidate region localization and ROI Pooling for classification.
- Model trained on MAFA and AFLW datasets, achieving a mean Average Precision of 0.73 across all classes.
- Highest accuracy achieved in detecting faces without masks, while the lowest accuracy observed for incorrectly masked faces.
- Addresses the gap in face detection by specifically targeting masked faces, offering potential applications in surveillance, healthcare, and public safety where mask compliance is essential.

[12]A Novel Single Shot Multibox Detector based Face Mask Detection

- Face mask detection using the Single Shot MultiBox Detector (SSD) framework is proposed as a method to accurately identify whether individuals are wearing masks in real-time scenarios.
- A dataset comprising images of masked and unmasked faces is collected and annotated, then utilised to train the SSD model, leveraging its ability for object localization and classification simultaneously.
- During the inference phase, the trained SSD model is applied to input images or video frames, analysing detected faces and assigning labels based on the presence or absence of masks, along with bounding box coordinates.
- Experimental evaluations on various real-world datasets demonstrate the effectiveness and efficiency of the approach, achieving high accuracy in face mask detection.
- This method offers a significant improvement in accuracy and efficiency compared to existing approaches, providing a robust solution for face mask detection in real-time scenarios.

[13]Face Masked and Unmasked Humans Detection and Tracking in Video Surveillance

- Face masks have become commonplace globally due to COVID-19, posing challenges for existing face detection and tracking systems.
- Proposed solution involves extracting facial features from top regions (eye, eyebrow, forehead) for improved detection and tracking.

- Methodology includes two models: a face detector trained on a joint dataset and a long-term object tracker using pre-trained YOLOv4 weights and DeepSORT.
- Face detection model achieves a testing accuracy of 93.33% and a loss of 26.92%, comparable to state-of-the-art models.
- Addresses the gap in current technology by focusing on top facial regions, offering potential for more effective detection and tracking despite face masks.

[14]Masked Faces Classification using Deep Convolutional Neural Network with VGG-16 Architecture

- Recent attention in object detection, notably face mask detection, underscores the importance of biometric technologies, especially amidst potential pandemics.
- Facial biometrics emerge as the most secure authentication method, prompting the need for advancements in face mask detection and classification.
- Existing technologies primarily cater to fair-skinned individuals, prompting a study to enhance performance on dark-skinned faces using a convolutional neural network with VGG-16 architecture.
- The evaluation of the system reveals improved performance, addressing a critical gap in existing solutions.
- This project aims to contribute to the enhancement of face mask detection and identification technologies, particularly for dark-skinned individuals, providing a more inclusive and robust solution than current systems.

[15]Masked Face Detection with Illumination Awareness

- Mask mandates have been widely implemented to mitigate COVID-19 transmission, prompting the development of vision-based approaches to monitor mask compliance in public settings.
- Challenges persist in vision-based masked face detection due to insufficient datasets covering variations in lighting, object scales, mask types, and occlusion levels.
- This paper investigates the effectiveness of a lightweight masked face detection system under different lighting conditions and explores enhancing its performance with an image enhancement algorithm and an illumination awareness classifier.
- A novel dataset of human subjects with and without face masks in various lighting conditions is introduced, facilitating the training of an illumination awareness classifier.
- Experimental results demonstrate that integrating the masked face detection system with illumination awareness and an image enhancement algorithm can significantly improve performance metrics such as Accuracy, F1-score, and AP-M by up to 8.6%, 7.4%, and 8.5%, respectively.

[16]Deep Learning Model based Face Mask Detection for Automated Mandation

- Utilises Deep Convolutional Neural Network (CNN) models to automate face mask detection.

- NasNet architecture demonstrated superior performance compared to eight other state-of-the-art models.
- Achieved 100% testing accuracy on Simulated Masked Face Dataset (SMFD) and 99.872% testing accuracy on Real-World Masked Face Dataset (RMFD).
- Addresses the challenge of efficiently determining whether an individual is wearing a mask or not.
- Provides a promising solution to enhance mask-wearing compliance and reduce the risk of COVID-19 transmission.

[17]Masked Face Recognition via Self-Attention Based Local Consistency Regularisation

- Existing face recognition systems face challenges due to the widespread use of masks during the COVID-19 pandemic, leading to heavy occlusions.
- The proposed system combines robust masked face detection and alignment using RetinaFace with a deep CNN network trained to recognize masked faces by minimising ArcFace loss and incorporating a local consistency regularisation (LCR) loss.
- The method aims to learn globally discriminative face representations while ensuring locally consistent representations between non-occluded faces and their masked counterparts.
- Experiments on the masked LFW dataset demonstrate superior performance compared to multiple state-of-the-art methods, showcasing the effectiveness of the proposed approach.
- The system is implemented on a portable Jetson Nano device, enabling real-time masked face recognition, thus addressing the practical need for reliable face recognition despite mask-wearing.

[18]Real-time Face Mask Detection System on Edge using Deep Learning and Hardware Accelerators

- Real-time face mask detection using AI, particularly with Object Detection models like YOLOv5s and YOLOv5l, is highlighted as an advanced method for detecting face masks and their wearing conditions in various settings.
- The system efficiently detects and classifies face masks based on their wearing condition, while also counting and storing the count in a CSV file format with timestamps, enhancing data tracking and management.
- Deployment on edge devices such as Nvidia Jetson Nano and Jetson Xavier NX enables real-time inference, emphasising the integration of Edge AI for practical application in different environments.
- Performance metrics such as mean Average Precision (mAP) and frames per second (fps) are provided, showcasing the effectiveness of the YOLOv5l model over YOLOv5s, and the superior hardware capability of Nvidia Jetson Xavier NX for real-time inference.

- The proposed project contributes to the existing literature by addressing the need for efficient and accurate real-time face mask detection systems, leveraging state-of-the-art deep learning models and edge computing technology. It offers advancements in accuracy, speed, and practical deployment, thereby presenting a notable improvement over existing methods.

[19]Machine Learning based Real-Time Face Mask Detection System

- Summarised research on face mask detection using basic ML packages in Python like TensorFlow, Keras, and OpenCV.
- Highlighted the importance of mask detection in controlling the spread of COVID-19 and its integration into public and government sectors.
- Described the method's accuracy in detecting faces and identifying mask presence based on two datasets.
- Addressed the surveillance aspect of the task for security and public awareness purposes.
- Positioned the proposed project within the context of existing literature, emphasising its contribution in achieving a high accuracy rate and its potential for real-time application in various scenarios.

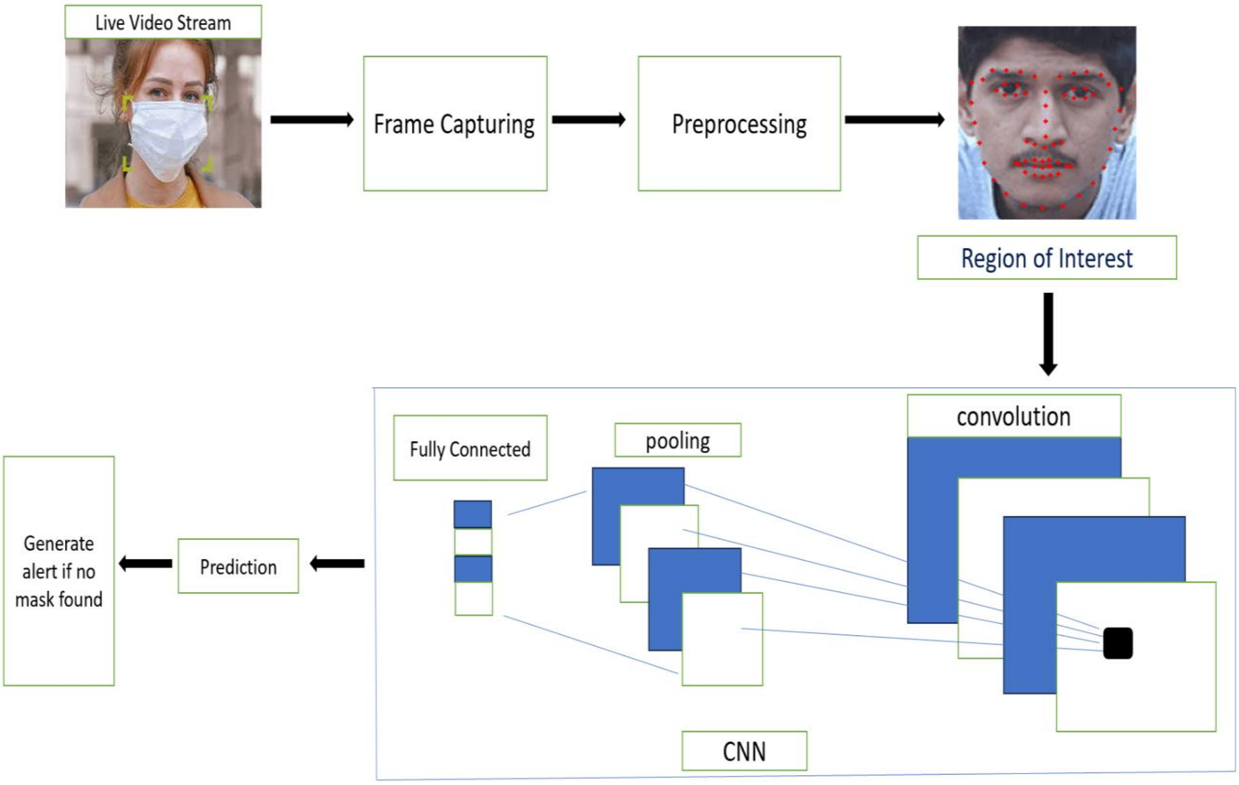
[20]A Real-Time Face Mask Detection Using SSD and MobileNetV2

- Utilisation of Convolutional Neural Networks (CNNs) for developing a face mask detection system amidst the COVID-19 pandemic.
- Integration of ResNet model for extracting multiple faces within a single image using SSD (Single Shot Multibox Detector), coupled with MobileNetV2 Architecture for mask detection.
- The proposed model boasts a 99% accuracy rate, outperforming other face recognition models.
- Incorporation of a dataset comprising hidden morphed masked images to enhance model accuracy.
- Emphasis on real-time application of the system for face mask detection in public places, citing the urgent need amid the COVID-19 crisis.

Problem Definition

Coronavirus disease is an infectious disease caused by the SARS-CoV-2 virus. Most virus-infected people will suffer from a mild to severe respiratory illness and recover without the need for special treatment. However, some people will get severe ailments and require medical attention. Elderly people and people with underlying medical conditions including cancer, diabetes, cardiovascular disease, or chronic respiratory problems are more prone to have serious illness. Anyone who contracts COVID-19 might get seriously sick or die at any age. Being knowledgeable about the illness and the virus's propagation is the greatest strategy to stop or slow down transmission. By keeping a distance of at least one metre between people, using a mask that fits properly, and often washing your hands or using an alcohol-based rub, you may prevent infection in both yourself and other people. When it's your turn, get your vaccination, and abide by any local advice. When an infected person coughs, sneezes, speaks, sings, or breathes, the virus can spread from their mouth or nose in minute liquid particles. Face masks are an excellent defence against any pandemic that may spread via the air, thus many public service providers have made it necessary for customers to wear them appropriately in order to use their services with every incidence of a new version of the illness emerging. Using deep learning tools like TensorFlow, Keras and OpenCV, our proposed work creates a streamlined method to accomplish this goal. At the entrances to manufacturing plant regions, the suggested model accurately recognises faces from video feeds and determines whether the individual is wearing a mask or not. We also created face recognition attendance system to avoid manual attendance. We also not used biometric because multiple users will be using it.

Architecture diagram



[1]Architecture Diagram for Face Mask Detection System

Image Preprocessing

The accuracy of a model depends on the dataset's quality. The initial data cleaning is done to get rid of the problematic photographs that were discovered in the dataset. In order to reduce the workload on the machine during training and to achieve the best results, the images are scaled to a predefined size of 96 by 96. After that, the images are categorised as having masks or not. The photo array is then transformed into a NumPy array to speed up calculation. Additionally, the preprocess input function of the MobileNetV2 is used. The training dataset is subsequently expanded and enhanced using the data augmentation technique. By using the ImageDataGenerator function and the required rotation, zoom, and horizontal or vertical flip parameters, numerous variations of the same image can be produced.

The training sample size has been increased to avoid overfitting. As a result, the generalizability and resilience of the trained model are enhanced. The full dataset is then split into training data and test data in an 8:2 ratio using random photos from the set. The stratify option is used in both the training and testing datasets in order to keep the same percentage of the data from the original dataset.

Arrays are made using images. Two new labels, or empty list data, have been added. After adding all of the photos to the array in data, label each associated image—with or without a mask—in labels.

The path to each image is included in the list of all images that os.listdir(path) returns. The image is also loaded using load image from the keras.preprocessing.image package, and target size is set to (224,224). Then, using the keras image to array function, the image may be converted into an array. Preprocessing is done on the photos. The image is added using data.append, and the category is added using labels. The labels are split into 0 and 1 using the Label Binarizer from sklearn.preprocessing.

np.array is used to convert a list into an array.

(train X,test X,train AND,test Y)

test_size=20%

train_size=80%

random_state=42

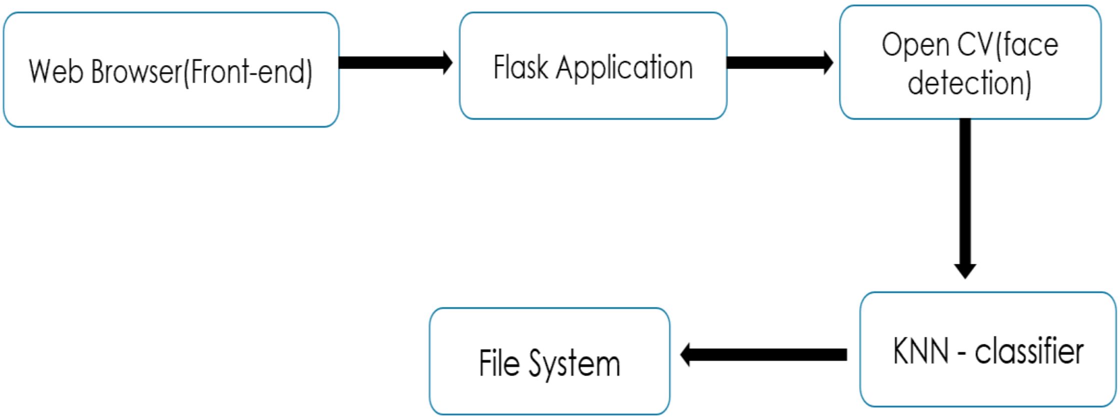
After the input image is processed as an array , it will send that into Mobilenet and then we will do max pooling, flatten it, fully connect and then give output.

0.0006 is the learning rate, epoches=25 and bs=32.

Two models are used: Mobile-net model and normal model. One is the head model and the other is the base model. Output from the Mobile-net model is passed to the normal model.

Feature Extraction

Start with a picture of a person who is not wearing a face mask in order to build a dataset of faces wearing them using facial landmarks. The location of the face within the bounding box of the image is then determined using face detection. We can extract the face Region of Interest once we know where the face is in the image (ROI). Then, using facial landmarks, we can recognise the mouth, nose, and other features of the face. Given that this mask is applied automatically, the facial landmarks will be used to decide where the mask should be placed on the face (namely, the points around the chin and nose). The mask is then scaled and twisted before being placed on the faces .



[2]Architecture Diagram for Face Recognition Attendance System

This diagram illustrates the flow of data and interactions between different components:

Web Browser: The user interacts with the system through a web browser, making HTTP requests (GET, POST) to the Flask web application.

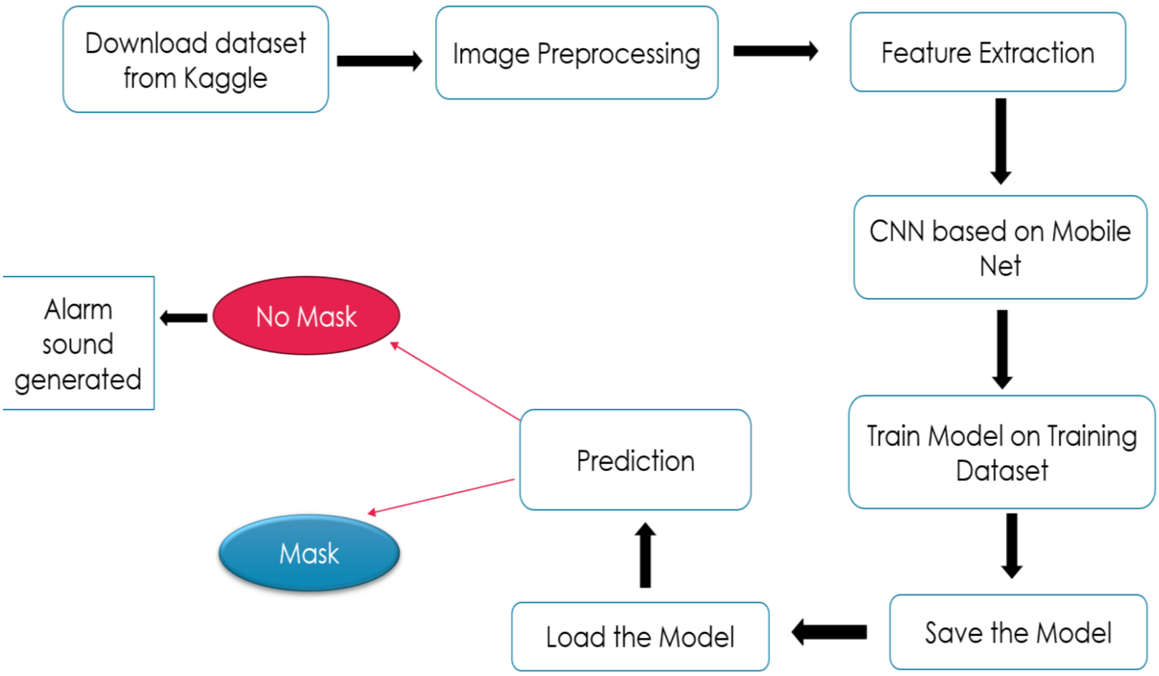
Flask App: The Flask web application handles incoming HTTP requests and interacts with various components of the system, including OpenCV for face detection, scikit-learn for face recognition, and the file system for storing images and attendance data.

OpenCV (face detection): OpenCV is used for real-time face detection from the video stream captured by the webcam. Detected faces are then passed to the face recognition component.

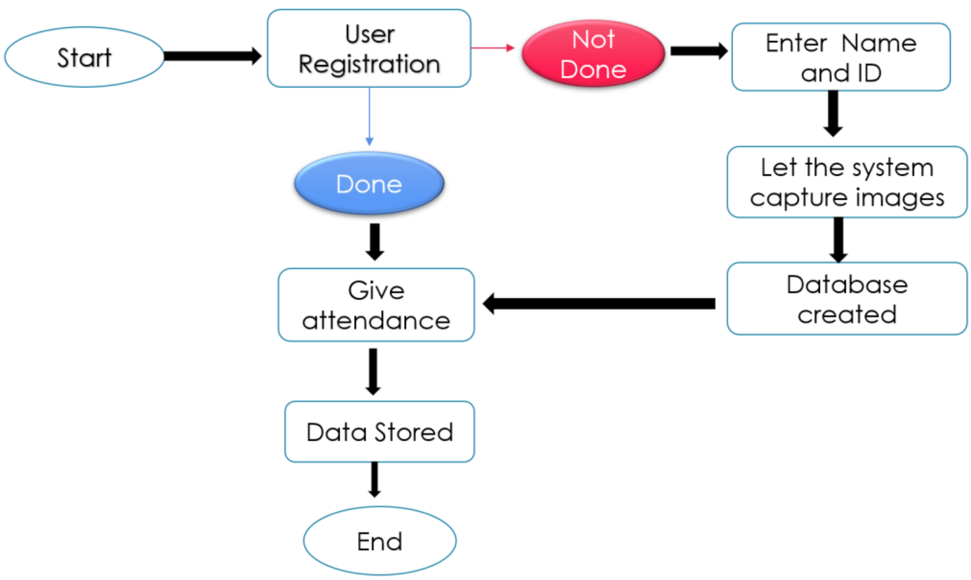
scikit-learn (KNN Classifier): scikit-learn is used to train a K-Nearest Neighbours (KNN) classifier for face recognition. The trained model is then used to recognize faces in the captured frames.

File System: The file system is used to store images of registered users, as well as attendance data in CSV format.

Flow Diagram



[3]Flow-chart for Face Mask detection System



[4]Flow-chart for Face Recognition Attendance System

Tables

Epochs	loss	accuracy	val_loss	val_accuracy
5	0.6913	0.5000	0.6888	0.5100
10	0.6902	0.5025	0.6872	0.5100
15	0.6890	0.5075	0.6857	0.5100
20	0.6882	0.5075	0.6841	0.5250
25	0.6873	0.5138	0.6826	0.6250

[1]Training Model on SGD Optimizer

Epochs	loss	accuracy	val_loss	val_accuracy
5	0.0346	0.9888	0.0261	0.9909
10	0.0161	0.9947	0.0258	0.9922
15	0.0149	0.9947	0.0362	0.9857
20	0.0125	0.9960	0.0171	0.9948
25	0.0165	0.9941	0.0180	0.9922

[2]Training Model on Adam Optimizer

Parameter	Detail
Learning Rate	0.0006
Epochs	25
Batch Size	32
Optimizer	Adam
Loss Function	binary_crossentropy

[3]Hyper Parameters used in Training

Module Description

Module 1: Face Mask Detection

This module will use a deep learning model to detect faces in real time and determine whether or not the person is wearing a mask. The model will be trained on a large dataset of images of faces with and without masks. The output of the model will be a binary classification, indicating whether or not a mask is detected.

Module 2: Alarm Generation

This module will generate an alarm sound if the face mask detection module detects that the person is not wearing a mask. Further, the temperature sensing module detects that the person has a fever or not.

Module 3: Face Recognition Attendance

This module will be developed using Python and Flask framework to automate the attendance marking process. This system employs facial recognition technology to identify individuals and record their attendance in various settings such as classrooms, offices, or events. By utilising computer vision algorithms, it eliminates the need for traditional manual attendance methods, saving time and resources while enhancing accuracy and efficiency.

Module 4: Registration in Attendance System

This module says that if the client is not registered, he/she can be given an opportunity to register first, then he/she can give attendance.

Sample codes:-

[a] For training of model - Real Time Face mask detection

```
from keras.preprocessing.image import ImageDataGenerator
from keras.applications.mobilenet_v2 import MobileNetV2
from keras.layers import AveragePooling2D
from keras.layers import Dropout
from keras.layers import Flatten
from keras.layers import Dense
from keras.layers import Input
from keras.models import Model
from keras.optimizers import Adam
from keras.applications.mobilenet_v2 import preprocess_input
from keras.preprocessing.image import img_to_array
from keras.preprocessing.image import load_img
from tensorflow.keras.utils import to_categorical
from sklearn.preprocessing import LabelBinarizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from imutils import paths
import matplotlib.pyplot as plt
import numpy as np
import os

INIT_LR = 0.0006
EPOCHS = 25
BS = 32

DIRECTORY = r"C:\Project\Face-Mask-Detection-master\dataset"
CATEGORIES = ["with_mask", "without_mask"]


print("[INFO] loading images...")

data = []
labels = []

for category in CATEGORIES:
    path = os.path.join(DIRECTORY, category)
    for img in os.listdir(path):
            img_path = os.path.join(path, img)
            image = load_img(img_path, target_size=(224, 224))
```

```
        image = img_to_array(image)
        image = preprocess_input(image)

        data.append(image)
        labels.append(category)


lb = LabelBinarizer()
labels = lb.fit_transform(labels)
labels = to_categorical(labels)

data = np.array(data, dtype="float32")
labels = np.array(labels)

(trainX, testX, trainY, testY) = train_test_split(data, labels,
        test_size=0.20, stratify=labels, random_state=42)


aug = ImageDataGenerator(
        rotation_range=20,
        zoom_range=0.15,
        width_shift_range=0.2,
        height_shift_range=0.2,
        shear_range=0.15,
        horizontal_flip=True,
        fill_mode="nearest")

baseModel = MobileNetV2(weights="imagenet", include_top=False,
        input_tensor=Input(shape=(224, 224, 3)))

headModel = baseModel.output
headModel = AveragePooling2D(pool_size=(7, 7))(headModel)
headModel = Flatten(name="flatten")(headModel)
headModel = Dense(128, activation="relu")(headModel)
headModel = Dropout(0.5)(headModel)
headModel = Dense(2, activation="softmax")(headModel)

model = Model(inputs=baseModel.input, outputs=headModel)
# loop over all layers in the base model and freeze them so they will
# *not* be updated during the first training process

for layer in baseModel.layers:
        layer.trainable = False
```

```
print("[INFO] compiling model...")
opt = Adam(lr=INIT_LR, decay=INIT_LR / EPOCHS)
model.compile(loss="binary_crossentropy", optimizer=opt,
	metrics=["accuracy"])

print("[INFO] training head...")
H = model.fit(
	aug.flow(trainX, trainY, batch_size=BS),
	steps_per_epoch=len(trainX) // BS,
	validation_data=(testX, testY),
	validation_steps=len(testX) // BS,
	epochs=EPOCHS)

# make predictions on the testing set
print("[INFO] evaluating network...")
predIdxs = model.predict(testX, batch_size=BS)

predIdxs = np.argmax(predIdxs, axis=1)

print(classification_report(testY.argmax(axis=1), predIdxs,
	target_names=lb.classes_))


print("[INFO] saving mask detector model...")
model.save("mask_detector.model", save_format="h5")

N = EPOCHS
plt.style.use("ggplot")
plt.figure()
plt.plot(np.arange(0, N), H.history["loss"], label="train_loss")
plt.plot(np.arange(0, N), H.history["val_loss"], label="val_loss")
plt.plot(np.arange(0, N), H.history["accuracy"], label="train_acc")
plt.plot(np.arange(0, N), H.history["val_accuracy"], label="val_acc")
plt.title("Training Loss and Accuracy")
plt.xlabel("Epoch #")
plt.ylabel("Loss/Accuracy")
plt.legend(loc="lower left")
plt.savefig("plot.png")
```

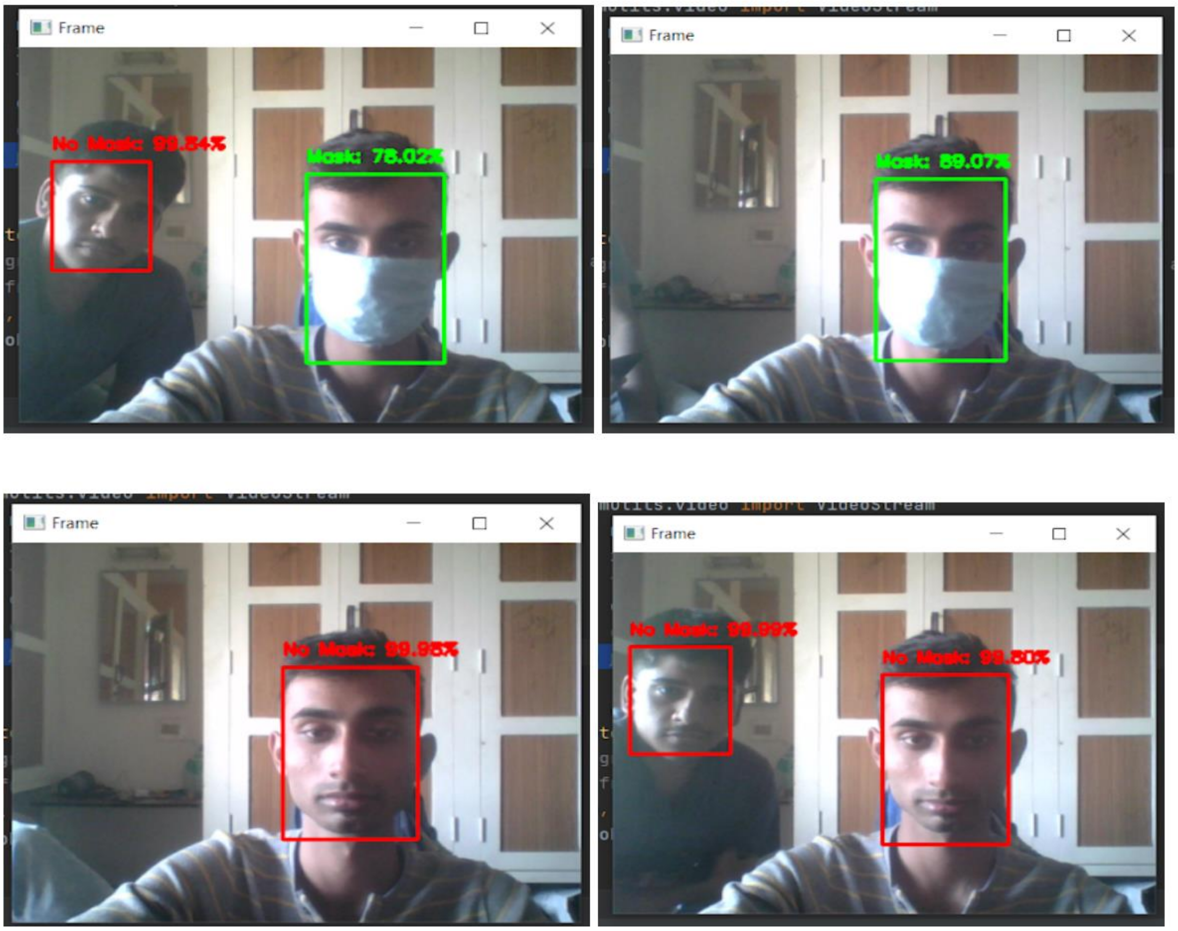
[b] For detection of the system

```
from keras.applications.mobilenet_v2 import preprocess_input
from keras.preprocessing.image import img_to_array
from keras.models import load_model
from imutils.video import VideoStream
import numpy as np
import imutils
import time
import cv2
import os
from playsound import playsound


def detect_and_predict_mask(frame, faceNet, maskNet):
    (h, w) = frame.shape[:2]
    blob = cv2.dnn.blobFromImage(frame, 1.0, (224, 224),
                                 (104.0, 177.0, 123.0))

    faceNet.setInput(blob)
    detections = faceNet.forward()
    print(detections.shape)
    faces = []
    locs = []
    preds = []


    for i in range(0, detections.shape[2]):
        confidence = detections[0, 0, i, 2]
        if confidence > 0.5:
            box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
            (startX, startY, endX, endY) = box.astype("int")
            (startX, startY) = (max(0, startX), max(0, startY))
            (endX, endY) = (min(w - 1, endX), min(h - 1, endY))
            face = frame[startY:endY, startX:endX]
            face = cv2.cvtColor(face, cv2.COLOR_BGR2RGB)
            face = cv2.resize(face, (224, 224))
            face = img_to_array(face)
            face = preprocess_input(face)
            faces.append(face)
            locs.append((startX, startY, endX, endY))

    if len(faces) > 0:
```

```
        faces = np.array(faces, dtype="float32")
        preds = maskNet.predict(faces, batch_size=32)

    return (locs, preds)

prototxtPath = r"face_detector\deploy.prototxt"
weightsPath = r"face_detector\res10_300x300_ssd_iter_140000.caffemodel"
faceNet = cv2.dnn.readNet(prototxtPath, weightsPath)


maskNet = load_model("mask_detector.model")


print("[INFO] starting video stream...")
vs = VideoStream(src=0).start()

while True:
    frame = vs.read()
    frame = imutils.resize(frame, width=400)
    (locs, preds) = detect_and_predict_mask(frame, faceNet, maskNet)
    for (box, pred) in zip(locs, preds):
        (startX, startY, endX, endY) = box
        (mask, withoutMask) = pred
        label = "Mask" if mask > withoutMask else "No Mask"
        color = (0, 255, 0) if label == "Mask" else (0, 0, 255)

        if label == "No Mask":
            playsound("ravi.wav")

            # s = Sound()
            # s.read('milky.wav')
            # s.play()


        label = "{}: {:.2f}%".format(label, max(mask, withoutMask) * 100)

        cv2.putText(frame, label, (startX, startY - 10),
                    cv2.FONT_HERSHEY_SIMPLEX, 0.45, color, 2)
        cv2.rectangle(frame, (startX, startY), (endX, endY), color, 2)


    cv2.imshow("Frame", frame)
    key = cv2.waitKey(1) & 0xFF
```

```
    # if the `q` key was pressed, break from the loop
    if key == ord("q"):
        break


cv2.destroyAllWindows()
vs.stop()
```

Sample Test Plans, Test Cases:



[5]Input output of the face mask detection system

Performance Metrics:

	precision	recall	f1-score	support
with_mask	1.00	0.99	0.99	383
without_mask	0.99	1.00	0.99	384
accuracy			0.99	767
macro avg	0.99	0.99	0.99	767
weighted avg	0.99	0.99	0.99	767

[6] Performance metric of face mask detection system

The various metrics used to evaluate the model are accuracy, precision, recall, and f1 score.

$$Accuracy = \frac{Tp + Tn}{Tp + Tn + Fp + Fn}$$

$$Precision = \frac{Tp}{Tp + Fp}$$

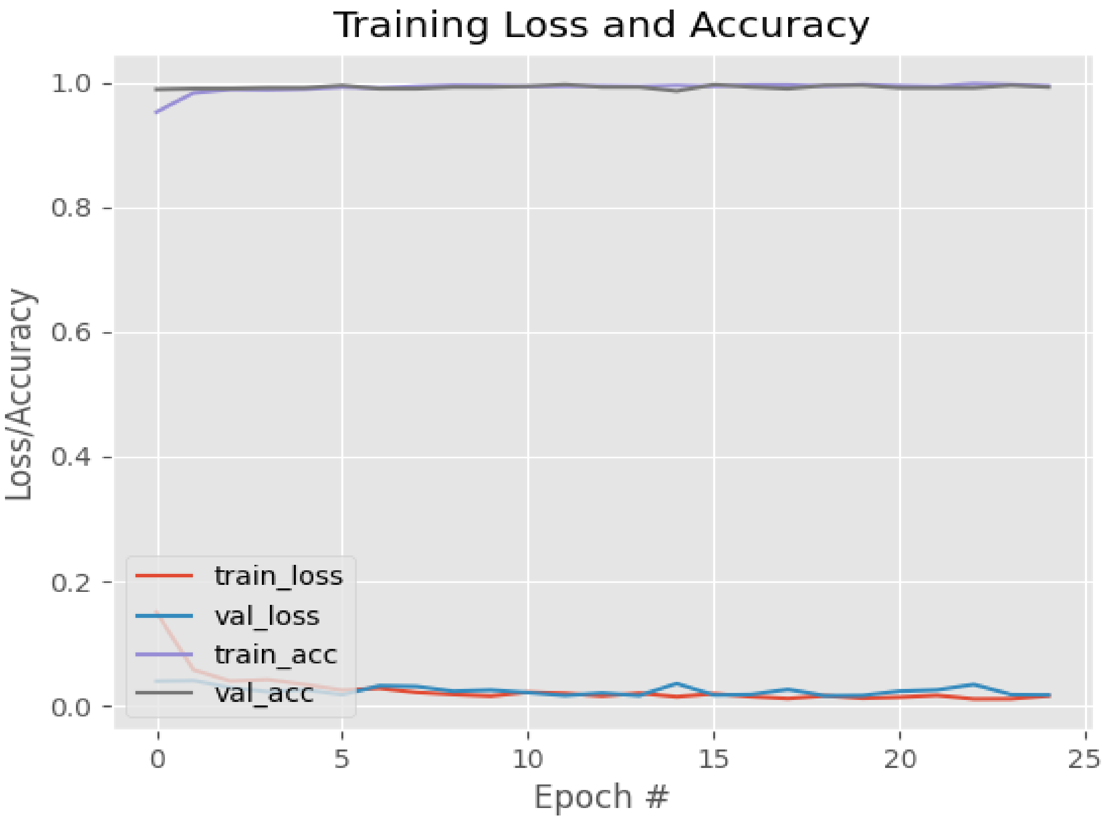
$$Recall = \frac{Tp}{Tp + Fn}$$

$$F1 = \frac{2 * (Precision * Recall)}{Precision + Recall}$$

In the above, *Tp* represents: True positive, *Tn*: True negative, *Fp*: False positive, and *Fn*: False negative.

True positives are accurately predicted as being in a positive class, whereas false positives are images that were incorrectly predicted as being in a positive class. *True negatives* are accurately predicted to be in the negative class, whereas false negatives are incorrectly predicted to be in the negative class.

The accuracy of the masked individual identified by the developed model provides a good standard of prediction. In Fig.7 below, which shows training accuracy, the model had struggled to acquire features until it reached 5 epochs, following which the curve remained steady. The accuracy was around 97% after 25 epochs. The green curve displays the training accuracy, while the aqua line gives the validation dataset. The training and validation loss curve, the red line represents loss in the training dataset, which is smaller than 0.1, and the loss in the validation dataset is represented by the blue line, which is also less than 0.2 after 5 epochs.



[7]Training Loss & Accuracy Graph

Model	Accuracy [%]	Time Per Epoch [s]	Model Size [MB]
MobilenetV2	97	10.16	10.9
DenseNet-121	98	10.94	96.7
VGG-19	95	10.34	79.4

[4] Comparison Table of different models

As can be seen in the table, different models were tested on the same data-set, giving an accuracy above 95%. The VGG-19 model had the lowest overall performance, down by 3% in accuracy from the proposed model.

On comparing several models with the proposed model based on accuracy, size, and training speed, we can find that DenseNet-121, for example, performs best as it has low training time and also has large memory. On the other hand, MobileNet-V2 performs marginally more inferiorly and is substantially slower to train but has a smaller memory.

Conclusion

In order to prevent the COVID-19 new variant from spreading, this research suggests a face mask identification system for real-time video that can detect whether a person is wearing a mask automatically (see Fig. 10). The suggested system can determine whether a face mask is present or absent by utilising Keras, OpenCV, and CNN. The model provides accurate and prompt results along with a warning sound. An accuracy of about 97% is produced by the trained model. For a real-time monitoring system, this methodology is a strong contender thanks to its accuracy and computing effectiveness. The suggested approach can be used in locations including airports, businesses, schools, retail centres, and train stations other than manufacturing plants.

Face recognition-based attendance systems represent the future of attendance tracking. With their accuracy, efficiency, and security features, they offer a superior alternative to traditional methods. However, it is essential to address concerns related to privacy and security while ensuring regulatory compliance. As technology evolves, we can expect even more exciting developments in this field.

Future Enhancement

- Provide visual or audio alerts to remind users to maintain safe distances.
- Combine face recognition with other modalities (voice, gait, iris) for heightened security.
- Allow users to choose additional factors for layered authentication.
- Incorporate a feedback mechanism for users to report false positives or negatives, which can be used to further refine and improve the face recognition algorithm.
- Develop a mobile application that communicates with the Flask server to allow students and teachers to check attendance records, receive notifications, and view statistics on their smartphones.

References

[1]A. S. Joshi, S. S. Joshi, G. Kanahasabai, R. Kapil and S. Gupta, "Deep Learning Framework to Detect Face Masks from Video Footage," 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN), Bhimtal, India, 2020, pp. 435-440, doi: 10.1109/CICN49253.2020.9242625.

[2]Z. Yu, X. Li, P. Wang and G. Zhao, "TransRPPG: Remote Photoplethysmography Transformer for 3D Mask Face Presentation Attack Detection," in IEEE Signal Processing Letters, vol. 28, pp. 1290-1294, 2021, doi: 10.1109/LSP.2021.3089908.

[**3]**Y. Chen et al., "Face Mask Assistant: Detection of Face Mask Service Stage Based on Mobile Phone," in IEEE Sensors Journal, vol. 21, no. 9, pp. 11084-11093, 1 May 1, 2021, doi: 10.1109/JSEN.2021.3061178.

[4]B. Wang, Y. Zhao and C. L. P. Chen, "Hybrid Transfer Learning and Broad Learning System for Wearing Mask Detection in the COVID-19 Era," in IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1-12, 2021, Art no. 5009612, doi: 10.1109/TIM.2021.3069844.

[5]A. Liu et al., "Contrastive Context-Aware Learning for 3D High-Fidelity Mask Face Presentation Attack Detection," in IEEE Transactions on Information Forensics and Security, vol. 17, pp. 2497-2507, 2022, doi: 10.1109/TIFS.2022.3188149.

[6]B. Kocacinar, B. Tas, F. P. Akbulut, C. Catal and D. Mishra, "A Real-Time CNN-Based Lightweight Mobile Masked Face Recognition System," in IEEE Access, vol. 10, pp. 63496-63507, 2022, doi: 10.1109/ACCESS.2022.3182055.

[7]S. Manzoor et al., "Edge Deployment Framework of GuardBot for Optimised Face Mask Recognition With Real-Time Inference Using Deep Learning," in IEEE Access, vol. 10, pp. 77898-77921, 2022, doi: 10.1109/ACCESS.2022.3190538.

[8]N. M. Alnaim, Z. M. Almutairi, M. S. Alsuwat, H. H. Alalawi, A. Alshobaili and F. S. Alenezi, "DFFMD: A Deepfake Face Mask Dataset for Infectious Disease Era With Deepfake Detection Algorithms," in IEEE Access, vol. 11, pp. 16711-16722, 2023, doi: 10.1109/ACCESS.2023.3246661.

[9]C. Yao, J. Ren, R. Bai, H. Du, J. Liu and X. Jiang, "Mask Attack Detection Using Vascular-Weighted Motion-Robust rPPG Signals," in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 4313-4328, 2023, doi: 10.1109/TIFS.2023.3293949.

[10]N. Kowalczyk, M. Sobotka and J. Rumiński, "Mask Detection and Classification in Thermal Face Images," in IEEE Access, vol. 11, pp. 43349-43359, 2023, doi: 10.1109/ACCESS.2023.3272214.

[11]I. A. Siradjuddin, Reynaldi and A. Muntasa, "Faster Region-based Convolutional Neural Network for Mask Face Detection," 2021 5th International Conference on Informatics and Computational Sciences (ICICoS), Semarang, Indonesia, 2021, pp. 282-286, doi: 10.1109/ICICoS53627.2021.9651744.

[12]N. Bhargava, P. S. Rathore, P. Vaishnav and A. Goswami, "A Novel Single Shot Multibox Detector based Face Mask Detection," 2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India, 2023, pp. 1028-1031, doi: 10.1109/ICAISS58487.2023.10250518.

[13]Y. Alayary, N. Shoukry, M. A. A. El Ghany and M. A. . -M. Salem, "Face Masked and Unmasked Humans Detection and Tracking in Video Surveillance," 2022 4th Novel Intelligent and Leading Emerging Sciences Conference (NILES), Giza, Egypt, 2022, pp. 211-215, doi: 10.1109/NILES56402.2022.9942375.

[14]O. Tolulope Ibitoye and O. Oluwole Osaloni, "Masked Facies Classification using Deep Convolutional Neural Network with VGG-16 Architecture," 2022 3rd International Conference on Electrical Engineering and Informatics (ICon EEI), Pekanbaru, Indonesia, 2022, pp. 168-171, doi: 10.1109/IConEEI55709.2022.9972288.

[15]T. H. Dinh, Q. M. Doan, N. L. Trung, D. N. Nguyen and C. -T. Lin, "Masked Face Detection with Illumination Awareness," 2022 IEEE 16th International Symposium on Medical Information and Communication Technology (ISMICT), Lincoln, NE, USA, 2022, pp. 1-6, doi: 10.1109/ISMICT56646.2022.9828233.

[16]P. Lokhande, S. Surati, H. Trivedi and B. Shrimali, "Deep Learning Model based Face Mask Detection for Automated Mandation," 2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT), Gharuan, India, 2023, pp. 166-170, doi: 10.1109/InCACCT57535.2023.10141795.

[17]D. Lin, Y. Li, Y. Cheng, S. Prasad and A. Guo, "Masked Face Recognition via Self-Attention Based Local Consistency Regularisation," 2022 IEEE International Conference on Image Processing (ICIP), Bordeaux, France, 2022, pp. 436-440, doi: 10.1109/ICIP46576.2022.9898076.

[18]S. Ruparelia, M. Jethva and R. Gajjar, "Real-time Face Mask Detection System on Edge using Deep Learning and Hardware Accelerators," 2021 2nd International Conference on Communication, Computing and Industry 4.0 (C2I4), Bangalore, India, 2021, pp. 1-6, doi: 10.1109/C2I454156.2021.9689421.

[19]S. Saranya, V. Surendar and P. V. Paul, "Machine Learning based Real-Time Face Mask Detection System," 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2022, pp. 1479-1483, doi: 10.1109/ICSCDS53736.2022.9760919.

[20]K. Balaji and S. Gowri, "A Real-Time Face Mask Detection Using SSD and MobileNetV2," 2021 4th International Conference on Computing and Communications Technologies (ICCCT), Chennai, India, 2021, pp. 144-148, doi: 10.1109/ICCCT53315.2021.9711784.